A Methodology and a Software Tool for Sensor Data Validation/Reconstruction: Application to the Catalonia Regional Water Network

Miquel À. Cugueró-Escofet[a], Diego García[a], Joseba Quevedo[a], Vicenç Puig[a], Santiago Espin[b], Jaume Roquet[b]

[a]*Supervision, Safety and Automatic Control Research Center (CS2AC), Polytechnic University of Catalonia (UPC), Terrassa Campus, Gaia Research Bldg. Rambla Sant Nebridi, 22. 08222 Terrassa, Barcelona, Spain (e-mail: {miquel.angel.cuguero,diego.garcia,joseba.quevedo,vicenc.puig}@upc.edu).*

[b]*ATLL Concessionària de la Generalitat de Catalunya, SA. Sant Martí de l'Erm, 30. 08970 Sant Joan Despí, Barcelona, Spain (e-mail: {sespin, jroquet}@atll.cat).*

Abstract

In this paper, a sensor data validation/reconstruction methodology applicable to water networks and its implementation by means of a software tool, are presented. The aim is to guarantee that the sensor data are reliable and complete in case that sensor faults occur. The availability of such dataset is of paramount importance in order to successfully use the sensor data for further tasks e.g. water billing, network efficiency assessment, leak localisation and real-time operational control. The methodology presented here is based on a sequence of tests and on the combined use of spatial models (SM) and time series models (TSM) applied to the sensors used for real-time monitoring and control of the water network. Spatial models take advantage of the physical relations between different system variables (e.g. flow and level sensors in hydraulic systems) while time series models take advantage of the temporal redundancy of the measured variables (here by means of a Holt-Winters (HW) time series model). First, the data validation approach, based on several tests of different complexity, is described to detect potential invalid or missing data. Then, the reconstruction process is based on a set of spatial and time series models used to reconstruct the missing/invalid data with the model estimation providing the best fit. A software tool implementing the proposed data validation and reconstruction methodology is also described. Finally, results obtained applying the proposed methodology to a real case study based on the Catalonia regional water network is used to illustrate its performance.

Keywords: Sensor Data Validation/Reconstruction, Fault Isolation, Model-Based Fault Diagnosis, Time Series

1. Introduction

Critical Infrastructure Systems (CIS), including water, gas or electricity networks, are complex large-scale systems geographically distributed and decentralized with a hierarchical structure. These systems require highly sophisticated supervisory and real-time control schemes, to ensure high performance achievement and maintenance when conditions are non-favourable [1, 2] due to e.g. sensor and actuator malfunctions (faults). Regarding the measurements in water systems, the commonly measured hydraulic and quality variables include water flow rate in links, pressure in nodes, water level in tanks, pH, conductivity and turbidity, as well as disinfectant and pollutant concentrations. For each measurement obtained from a sensor, the data (signals) are usually represented in the form of one dimensional time

series. Each sensor measures a physical quantity and converts it into a signal that can be read by the appropriate instrumentation. Then, the measuring system converts the sensor signals into values aiming to represent a certain real physical quantity. These values, known as raw data, need to be validated before further use in order to assure the reliability of the results derived from their usage. In systems like CIS, a telecontrol system is acquiring, storing and validating data gathered from different kind of sensors every given sampling time (e.g. every few minutes) to accurately real-time monitor the whole system. In the data acquisition process, several problems can occur, as those related with the communication system (e.g. between sensors and data loggers or in the telecontrol system itself) or outliers, producing lost or corrupted data which may be of great concern in order to have valid historic records. When this is occurring, lost data should be replaced by a set of estimated data which should be representative of the data lost, since missing data may severely jeopardise further processes needing complete datasets in order to get meaningful conclusions/analysis. Another common problem in CIS is caused by the unreliable sensors, which may be affected by faults e.g. offset, drift, freezing in the measurements [3, 4, 5]. These unreliable data should also be detected and replaced by forecasted data, since it may be used for system management tasks e.g. maintenance, planning, investment plans, billing, security and operational control [6] and system fault detection and isolation (Figure 1). In the case of water network applications, this system fault diagnosis may include e.g. network leaks isolation, as considered in [7, 8, 9]. However, the methodology presented here may well be applied to different applications involving a telemeasured sensor network, such as smart buildings or environmental systems (see, e.g. [10, 11]). In addition to the possible measurement deviations related to the sensor performance itself, the errors can also occur due to heterogeneous reasons, e.g. sensor installation, calibration or electrical problems. Thus, it is important to provide the data system with procedures that can detect these problems and assist the user in the monitoring and the processing of the incoming data. The data validation is an essential step to improve data reliability. Sensor data validation and reconciliation have been intensively addressed using least-squares approaches including Kalman filters (see, e.g. [12, 13]). These have been also used for data forecasting, as pointed out in [14], where a review of techniques for prediction of consumption in water and natural gas grids is presented. The basic idea of Kalman filter based methods in data validation and reconciliation is to allow gross error detection and to provide reconstructed data that is consistent with model/balance equations describing the system operation. The approach presented in this paper aims to assess the validity of each single sensor measurement by means of a set of tests exploiting not only the model equations (spatial redundancy) but also temporal redundancy, using time series models and a bank of low-level tests (non-model based) aiming to label the data with a certain quality index. Traditionally, data validation process has been developed by manual data analysis, performed by experienced users with the only assistance of basic data analysis and visualisation tools [15], which significantly limits the amount of data to be validated [16] and the abnormal situations which may be correctly detected [17]. However, the volume of real data acquired in CIS is dramatically increasing due to the increment of automated measurement systems allowing their monitoring [18]. Also, real-time operation, paramount in many real applications, makes human data validation even harder to pursuit. In order to cope with this situation and increase the reliability of the data diagnosis system, automatic data validation tools have arised e.g. NIKLAS for real and non-real time diagnosis of meteorological data [19]. Also, in [20] a data validation module is considered in the framework of an on-line water quality fault tolerant control system. Over the last 15 years, more and more affordable on-line sensors have become available, leading to ever increasing acceptance of on-line water monitoring [21]. These on-line systems allow to deploy control mechanisms that are optimized for and respond to the actual process conditions. However, on-line systems require a data validation method that is applicable to real-time incoming data. The major difference between on-line and off-line data validation lies in the available information and the required execution time. In contrast with the on-line execution, the off-line operation has no time restrictions because real-time constraints do not apply, and regarding the information, the whole set of data is available. Moreover, on-line data validation is usually required by a further real-time control system and thus the data are used for decision support (or decision making) just after being obtained. Consequently, the on-line data validation process should have low execution time, whereas the off-line data validation does not have this requirement.

According to the nature of the available knowledge, different kinds of data validation approaches may be considered, with varying degrees of sophistication. In general, one may distinguish between elementary signal-based ("low-level") methods and model-based ("high-level") methods (see, e.g. [6]). Elementary signal based methods use simple heuristics and limited statistical information of a given sensor [22]. Typically, these methods are based on validating either signal values or signal variations. On the one hand, in the signal value-based approach, data are assessed as valid or invalid according to two different thresholds (high and low) so data are assumed to be invalid when lying

outside these threshold values. On the other hand, methods based on signal variations look for high variations (peaks in the curve) and low variations (flat curve) in the signals. Model-based methods rely on the use of models to check the consistency of sensor data [21]. This consistency check is based on computing the difference between the predicted value from the model and the real value measured by the sensors. Then, this difference (known as residual) is compared with a threshold value (zero in the ideal case). When the residual is bigger than the corresponding threshold, a fault is assumed in the sensor; otherwise, the sensor is assumed to work properly. Moreover, the information of all the available residuals and models allows performing fault isolation in order to discover the faulty sensor. Models are usually derived using either multivariate procedures exploiting the correlation or analytical relations between several variables, sometimes measured at different times ("temporal redundancy") and/or locations ("spatial redundancy").

In this paper, a methodology is developed for validation and reconstruction of sensors data in a water network, taking into account not only spatial models (SM) but also time series models (TSM) for each flow and level meter here. Also, internal models of every component in the local equipment units (e.g. pumps, valves, flows, levels) are considered. SM take advantage of the relation between different variables in the system (e.g. demand, pump flows and tank levels) while TSM take advantage of the temporal redundancy of the measured variables, by means of Holt-Winters (HW) time series models [23]. Moreover, after the corrupted sensor data are detected, they must be replaced by adequate estimated data using the available temporal/spatial redundancy. The methodology is mainly applied to flow and level meters, since it exploits the temporal redundancy of flow and level data in a water network. In this paper, an operative software tool implementing the presented methodology which is able to properly handle raw sensor data (including storage, querying and visualization) is also presented. The proposed approach and tool are applied to several subsystems in the Catalonia regional water network (Figure 8) using raw data collected from *ATLL Concessionària de la Generalitat de Catalunya, SA* (ATLL), the company managing this water network.

The structure of the paper is as follows: In Sections 2 and 3, the methodology to validate/reconstruct the sensor data, in order to provide a reliable dataset when faulty situations occur within the sensor set, is proposed. In Section 4, the software tool implementing the proposed methodology is introduced. In Section 5, the application case study is presented, based on the Catalonia regional water network. The sensors in this network measure several real magnitudes of interest such demand and input tank flows and levels, considering real-world scenarios. Also, the corresponding results obtained applying the proposed methodology are detailed in Section 5. Finally, conclusions of this work are outlined in Section 6.

2. Proposed Methodology

2.1. Description

In real water networks such as the one considered here, there is usually a telemeasurement system acquiring, recording and validating data gathered from different kind of sensors at each sample time to accurately real-time monitor the whole network [6]. As discussed in the introduction, in this data acquisition process, problems in the communication system (e.g. between sensors and data loggers) or in the telemeasurement system itself (e.g. sensors may be affected by e.g. offset, drift or freezing faults), often arise and produce data loss, which may be of great concern in order to have valid historic records. These unreliable data should be detected and replaced by estimated data before they can be used for system management tasks such as maintenance, planning, billing and operational control, as depicted in the procedure in Figure 1. The input to this procedure is the raw data y_{raw} gathered from the sensors. The process is divided in two different stages: the first stage is related with the data validation, while the second stage addresses the reconstruction of invalid/missing data, before the data are stored in an operational database (DB) for further use. At the first stage (data validation, detailed in Figure 2), if the datum $y_{raw}(k)$ at a certain sample time k is validated, flag v is set to 1 and datum $y_{val}(k) = y_{raw}(k)$ is stored in the aforementioned operational DB as validated data. Conversely, if the datum $y_{raw}(k)$ is invalidated, flag v is set to 0 and the datum reconstruction process (second stage) is started, in order to provide a reconstructed estimation $y_{rec}(k)$ of the invalid/missing data $y_{raw}(k)$ to be stored in the DB. The whole procedure is further detailed in Algorithm 1 for the data validation stage and in Algorithm 2 for the data reconstruction stage. Here, communication and sensor faults are considered as faults affecting the telemeasurement system and the sensors, respectively, and the data detection/reconstruction procedure is used as a prefilter to estimate the invalid/missing data when these type of faults are occurring.

As discussed in the introduction, different types of data detection methods with distinct degrees of complexity may be considered according to the available system knowledge. This is the approach that the proposed methodology here

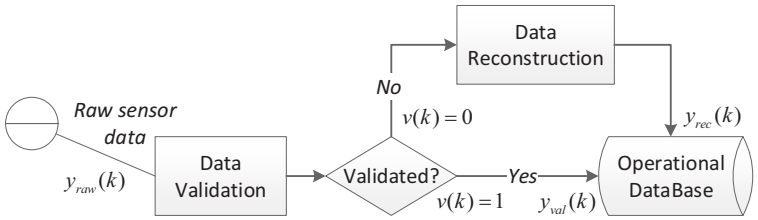

Figure 1. Raw data validation/reconstruction procedure

will follow. Generally, two types of methods are considered, one for elementary 'low-level' signal-based methods and another for 'high-level' model-based methods. The first type uses simple heuristics and limited statistical information from the sensors [22] [15] and is typically based on checking either signal values or variations, whilst the second type uses models for consistency-checking of the sensor data [21].

2.2. Validation Tests

The data detection process presented is inspired by the Spanish AENOR-UNE norm 500540 developed for data validation in meteorological stations [6]. The methodology presented here applies a set of consecutive detection tests to a given dataset (Figure 2), to finally assign a certain quality level q depending on the number of tests passed. Also, the corresponding tests passed are characterized by a validation vector $\mathbf{l}$, as shown in Figure 2. If the datum $y_{raw}(k)$ at a certain sample time k is voided at any validation level, flag v is set to 0 and the datum reconstruction process (second stage) is started. Conversely, if the datum $y_{raw}(k)$ pass all the validation levels, flag v is set to 1 and the data are validated (i.e. $y_{val}(k) = y_{raw}(k)$). In the latter situation i.e. validated datum $y_{raw}(k)$, $q(k) = 6$ and $\mathbf{l}(k) = [1, 1, 1, 1, 1, 1]$.

The validation tests include a set of 'low-level' tests (Levels 0 to 3, included) which check elementary signal properties, and a set of 'high-level' tests (Level 4 and Level 5), which rely on the use of models to check the consistency of the sensor data. The latter models are also used in the reconstruction stage of the potentially invalidated data, as explained in Section 3.3. As introduced in the last paragraph, if any of the validation tests in Figure 2 is not satisfied, $v = 0$. The validation procedure is also detailed in Algorithm 1.

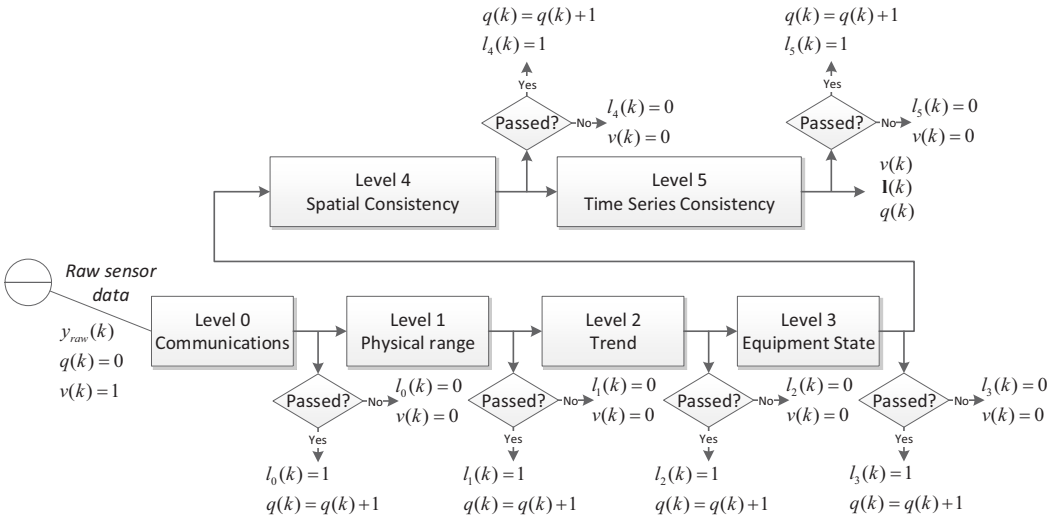

Figure 2. Data Validation Tests

An explanation of the test applied to each level is given next:

- **Level 0:** This level, also called *communications* level, checks whether the data are properly recorded at a regular sample rate by the acquisition system. If this is not fulfilled, there is some communication problem involving e.g. the data transmission from the ground sensors to the operational database. Hence, this level allows detecting problems in the data acquisition or communication system, which is one of the most common faults affecting telemeasurement systems, as the one considered here.

- **Level 1**: This level, also called *physical range* level, checks whether the data are within the physical range of the sensor acquiring the corresponding measurement. The expected range of the measurements may be obtained from e.g. sensor specifications or historical records of the data.

- **Level 2:** This level, also called *trend* level, checks whether the data derivative i.e. the magnitude change of the data among consecutive sample times, are within their expected rate. This allows detecting unexpected and possibly undesired sudden changes in the data, e.g. in a water network, tank water level sensors measurements cannot change more than several centimeters per minute.

- **Level 3:** This level, also called *equipment state* level, allows to check the consistency of the variables in a given equipment unit i.e. sensor or actuator. For example, in a water network system, in a pipe with a valve and a flow meter installed, there is a relation between the valve state and the flow meter reading.

- **Level 4:** This level, also called *spatial consistency* level, checks the consistency of the data collected by a certain sensor with its SM [24], i.e. the correlation between data coming from spatially-related sensors. This SM is obtained from the physical relations among these variables. In hydraulic systems, this relation is generally obtained from the mass balance model of the element relating the different measured variables involved.

- **Level 5:** This level, also called *time series consistency* level, checks for temporal consistency of a given sensor measurement, by means of a TSM obtained from sensor historical records under faultless assumption ([6]). A common method for time series signal forecasting is the HW approach ([23], [25]) because of its simplicity and low computational and storage requirements. In contrast to spatial consistency level, time series consistency level only uses information of the considered sensor without needing additional information (e.g. network topology or extra measurements from the system) to perform the validation, which makes it convenient when there is no such additional information available or the sensors needed by the corresponding spatial consistency level are unreliable. At this level, the analysis of the historic measurement records of a certain sensor are used to obtain the corresponding HW TSM sensor model and to validate the current data acquired by this element.

3. Model-based Data Validation/Reconstruction Levels

3.1. Models for Data Validation/Reconstruction

Model-based data validation/reconstruction relies on using models that exploit the temporal or spatial redundancy existing among the sensors. On the one hand, SM takes advantage of the relation between different variables physically related within the system. In water networks, this relation is generally obtained from the mass balance relating the different measured variables involved in a particular hydraulic element. For example, in a water tank (see Figure 3) the corresponding SM level estimation may be stated

$$\hat{x}_{SM}(k) = x(k-1) + \frac{\Delta t}{A}(q_{in}(k-1) - q_{out}(k-1)), \tag{1}$$

where $\hat{x}_{SM}$ is the spatial model tank level estimation, x is the measured tank level, q_{in} is the incoming tank flow, q_{out} is the outcoming tank flow and Δt is the sampling time, respectively. Estimation of other variables (e.g. $\hat{q}_{in}$, $\hat{q}_{out}$) may be obtained in a similar manner.

Real elements include uncertainty (e.g. due to noise, inaccuracy of the model, etc.) which may lead to the non-satisfaction of the mass balance in the element considered. Hence, consistency of the data collected by a certain sensor with its SM [24] (i.e. the correlation between data coming from spatially-related sensors) should take this uncertainty into account.

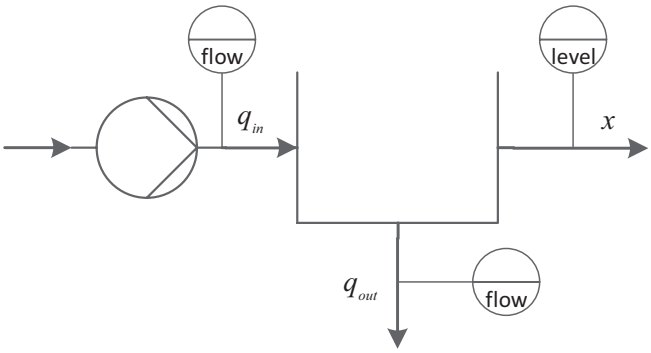

Figure 3. Single tank system schematics with single input and single demand

Alternatively, TSM takes advantage of the temporal redundancy of the measured variables. A wide used method for time series modelling because of its simplicity, low computational and storage requirements and ease of automation, is the HW approach [25]. This method, which was originally created for sales demand forecasting, has been used in a broad range of applications since its appearance. Exponential methods are first introduced in [26], where decreasing series of exponential weights are used. In [25], the former method is extended to include trend and seasonality terms. In [27, 28] multiple (i.e. double and triple) seasonality is explored, expanding the initial single seasonality expression of the former HW method, designed to cope with the sales demands monthly variations across a year period. Further alternative approaches to exponential smoothing forecasting may be found in [29] and [30]. Some issues of interest regarding its performance include the effect of the outliers in the forecasting, the consideration of the aforementioned different seasonal periods which may characterise the corresponding time series data sequence to be modelled (e.g. sales demands, water demands) or the consideration of prediction intervals which may provide reliability to the forecast. Regarding outliers, which may be produced by unexpected component behaviors (e.g. sensor malfunctions) these may degrade the performance of the HW method if not accommodated. In [31], this problem is considered and a robust version of the HW method against the outliers is presented, by recursively filtering their effect in the data main stream and applying the standard HW approach to the obtained filtered data. The latter approach is also considered here to provide robustness against the outliers. Moreover, there are different versions of the HW method e.g. additive or damped trend, additive or multiplicative seasonality, single or multiple seasonality [23]. Here, good performance has been attained with the additive single seasonality version, which estimated value is obtained for a forecasting horizon ℓ

$$\hat{x}_{TSM}(k) = \bar{R}(k-\ell) + \ell\bar{G}(k-\ell) + \bar{S}(k-L), \tag{2}$$

where $\bar{R}$ is the level estimation removing seasonality,

$$\begin{aligned}\bar{R}(k-\ell) =& \alpha\left(x(k-\ell) - \bar{S}(k-\ell-L)\right)\\ &+ (1-\alpha)\left(\bar{R}(k-\ell-1)\right.\\ &\left.+\bar{G}(k-\ell-1)\right),\end{aligned} \tag{3}$$

$\bar{G}$ is the trend estimation,

$$\begin{aligned}\bar{G}(k-\ell) =& \beta\left(\bar{R}(k-\ell) - \bar{R}(k-\ell-1)\right)\\ &+ (1-\beta)\,\bar{G}(k-\ell-1),\end{aligned} \tag{4}$$

S^- is the seasonal component estimation,

$$\begin{aligned}\bar{S}(k-\ell) =& \gamma\left(x(k-\ell) - \bar{R}(k-\ell)\right) \\ &+ (1-\gamma)\,\bar{S}(k-\ell-L),\end{aligned} \tag{5}$$

and L is the season (daily) periodicity, α, β and γ are the HW parameters (level, trend and season smoothing factors, respectively), x is the measured value and $\hat{x}_{TSM}$ is the TSM estimated value. The parameters α, β and γ are in the interval [0, 1] and can be estimated from historical data using the least-squares approach. Hence, analysing the historic records of a certain sensor, a HW TSM model can be obtained and used to estimate missing data of this element when a fault is affecting its readings.

3.2. Data Validation

On the one hand, the test check for the so-called 'low-level' tests are straightforward, since they rely on basic signal-based heuristics. On the other hand, the 'high-level' model-based tests rely on checking for consistency by means of the residuals $r_i(k)$, obtained from the difference between the system measurements and the corresponding SM or TSM estimations, expressed in input-output regressor form

$$r_i(k) = x_i(k) - \hat{x}_i(k) = x_i(k) - \phi_i^T(k)\theta_i, \tag{6}$$

where θ_i are the nominal parameters obtained using a training dataset, x_i is the sensor measurement, $\hat{x}_i$ is the model prediction and $\phi_i(k)$ is the regressor vector of dimensions $n_{\theta_i} \times 1$ including inputs $(u_i(k), u_i(k-1), u_i(k-2), ...)$ and outputs $(y_i(k), y_i(k-1), y_i(k-2), ...)$. The particular models used to compute the prediction $\hat{x}_i$ at instant k depend on the validation level considered (i.e. model-based level 4 or 5 in Figure 2, respectively), and are introduced in Section 3.3. Considering the uncertainty (e.g. modelling errors, noise), the detection test involves checking the condition

$$|r_i(k)| < \tau_i, \tag{7}$$

where τ_i is the detection threshold. The detection threshold can be determined using statistical methods [32] or set-membership approaches [33]. In the case of statistical methods, the noise is assumed to follow a normal distribution with known mean value μ_i and standard deviation σ_i [34]. Then, the threshold of the i-th residual can be determined as follows: $\tau_i = \mu_i + 3\sigma_i$, including the 99.7 % of the values of a normal distribution according to the *3-sigma* rule. Alternatively, when using a set-membership approach the noise is assumed to be unknown but bounded, with a priori known bound. Then, the threshold can be obtained by propagating the uncertainty to the residual computation [33]. Using either one or the other approach, the threshold in (7) is determined to include the values of the whole residual distribution in the faultless situation and hence, it may be used for fault detection purposes. This threshold is also useful to provide prediction interval bounds for the data forecasting process, so test condition (7) can be equivalently expressed as follows

$$x_i(k) \in [\underline{\hat{x}}_i(k), \bar{\hat{x}}_i(k)], \tag{8}$$

where $\bar{\hat{x}}_i(k) = \hat{x}_i(k) + \tau_i$ and $\underline{\hat{x}}_i(k) = \hat{x}_i(k) - \tau_i$, respectively. Condition (8) applies both to SM (1) and TSM (2) models. These interval bounds (8) consider the corresponding model behavior under faultless conditions including the uncertainty effect, as introduced in the residual bound condition (7). Hence, these bounds could alternatively be used in the data validation process, in order to decide whether a data sample at time instant k is reliable. The whole data validation process is detailed in Algorithm 1.

3.3. Data Reconstruction

As introduced in Section 2, when a fault is detected at the validation stage and the corresponding data are voided, a reconstruction process is started until the sensor data are validated again. The output of the data validation process (Figure 1) is used to identify the invalidated data that should be reconstructed. SM, related with Level 4 in Figure 2, and TSM, related with Level 5 in Figure 2, are used for this purpose, depending on the performance of each model. This data reconstruction process is detailed in Algorithm 2. The performance of each model is measured by the Mean Squared Error (MSE), evaluated in a moving horizon window

$$MSE(k) = \frac{1}{m} \sum_{j=k-m}^{k} e(j)^2, \tag{9}$$

where m is the number of data samples considered in the window, $e(j) = x(j) - \hat{x}(j)$ is the error at instant j, $x(j)$ is the measured value at instant j, $\hat{x}(j)$ is the estimated value by the model (SM or TSM, respectively) at instant j and k is the actual time instant. The model having best MSE index before the fault occurs (i.e. when the data validation process is not satisfactory) is used to produce the reconstructed sensor signal.

In order to produce the forecasted signal, it is desirable to use measured data instead of estimated data, to avoid model uncertainty effects in the forecasted value. This calls for the computation of $\hat{x}_i(k)|_\ell$ using (2) with $\ell \neq 1$ when possible and the usage of the different models obtained in a gain-scheduling fashion when e.g. the data are invalidated for more than a single time instant. HW TSM models may obtain forecasted values for different prediction horizons ℓ by design, if forecasted value at time k in (2) is rewritten as follows

$$\hat{x}_{TSM}(k)|_\ell = \bar{R}(k) + \ell \bar{G}(k) + \bar{S}(k - L + \ell), \tag{10}$$

Then, measured values may be used to produce the TSM forecasted signal within a complete season (day) L without using old forecasted values. In order to achieve this, the complete set of models (i.e. the models for each step within the complete season L) must be obtained at the calibration stage under faultless assumption, i.e. a HW TSM model $[\alpha_\ell, \beta_\ell, \gamma_\ell]$ may be obtained for $\ell = 1, \cdots, L$.

Similar procedure may be used in the same case study for alternative applications not related with data validation/reconstruction, as e.g. consumer demand prediction, in order to forecast water network user behavior beforehand. HW TSM models are specially suited to this end, since they were created in order to predict market product sales evolution according to consumer periodical behaviors [25], and user water consumption in district metered areas have a similar behavior.

4. Software Framework

The architecture of the software framework implemented is depicted in Figure 4. There are two main components: the *Data Management Web* application and the *Validation and Reconstruction* tool[1].

On the one hand, the Data Management component is a web application focused on collecting and serving time series data, i.e. observations coming from any kind of sensor. It allows authorized users to upload new data, download historical data and visualize data from anywhere using a device with a browser and Internet connection. Thus, this web-based data repository is highly available and provides a solution to the data-driven users to keep centralized data from different projects and sources. It also avoids typical datasets-usage related drawbacks e.g. data loss, sparse and duplicated data locations and emails with large datasets between project members.

On the other hand, the Validation and Reconstruction component allows users to apply the methodologies described in Sections 2 and 3 on data provided by the Data Management web application.

[1] Both software tools are proprietary software.

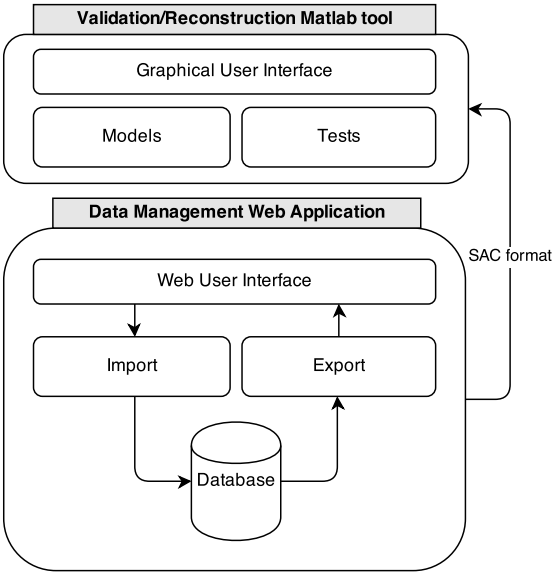

Figure 4. Software architecture diagram

4.1. Data Management Web Application

This module provides a user-friendly tool allowing to import and export data so that stored data are available to registered users with read permissions on the dataset. This point is important in order to respect existing confidential agreements: a user must have explicit permission on a dataset to be able to access or visualize it. Only the dataset's owner and the administrator can grant read permissions.

People working with data usually need to collect and prepare the raw data, e.g. remove outliers and fill missing data, before being able to apply further analysis e.g. statistical, exploratory or even to focus on the real objective of working with the corresponding data. These sort of tasks are time-consuming: there are many situations when the first two steps introduced take the 80 % of the whole data treatment process time. Thus, this tool provides three services in order to focus the efforts on the data themselves and not on how to collect, obtain and prepare them. These services are the following: the data import service, the data export service and the visualization service.

The data import service handles the data ingestion from different file formats (e.g. CSV, Excel, Access). An import wizard allows the user to specify the input data format, allowing the data to be loaded into the database after being specified. The data export service handles the data extraction. The user can specify the time period to export and the output file format. The current version of the tool allows to download data in CSV, Excel and SAC format[2]. Finally, the data visualization service provides a tool to visually explore the collected data. Hence, the user can plot multiple signals (e.g. time series) to do some exploratory analysis before downloading and to select only the relevant data. The visualization tool allows zooming and panning the time series.

This web application is implemented in two layers, a back-end (server layer) that handles the data storage and access with an underlaying data model, and a front-end (visual layer) to provide a friendly web-based user interface to interact with the three services described before. The back-end is developed with the Django[3] web framework connected to a database based on PostgreSQL. The front-end is implemented in HTML and JavaScript (see Figure 5). The Import and Export modules handle the operations of saving and querying data against the PostgreSQL Database server.

4.2. Validation and Reconstruction Matlab Tool

The Validation and Reconstruction methodologies, detailed in Sections 2 and 3, are summed up in Algorithm 1 and Algorithm 2, respectively. These methodologies are implemented in a software tool developed in Matlab. Matlab

[2]SAC format is a binary Mat-file containing a defined data structure.

[3]Django is a free open source web framework. Its primary goal is to facilitate the creation of complex, database-driven websites.

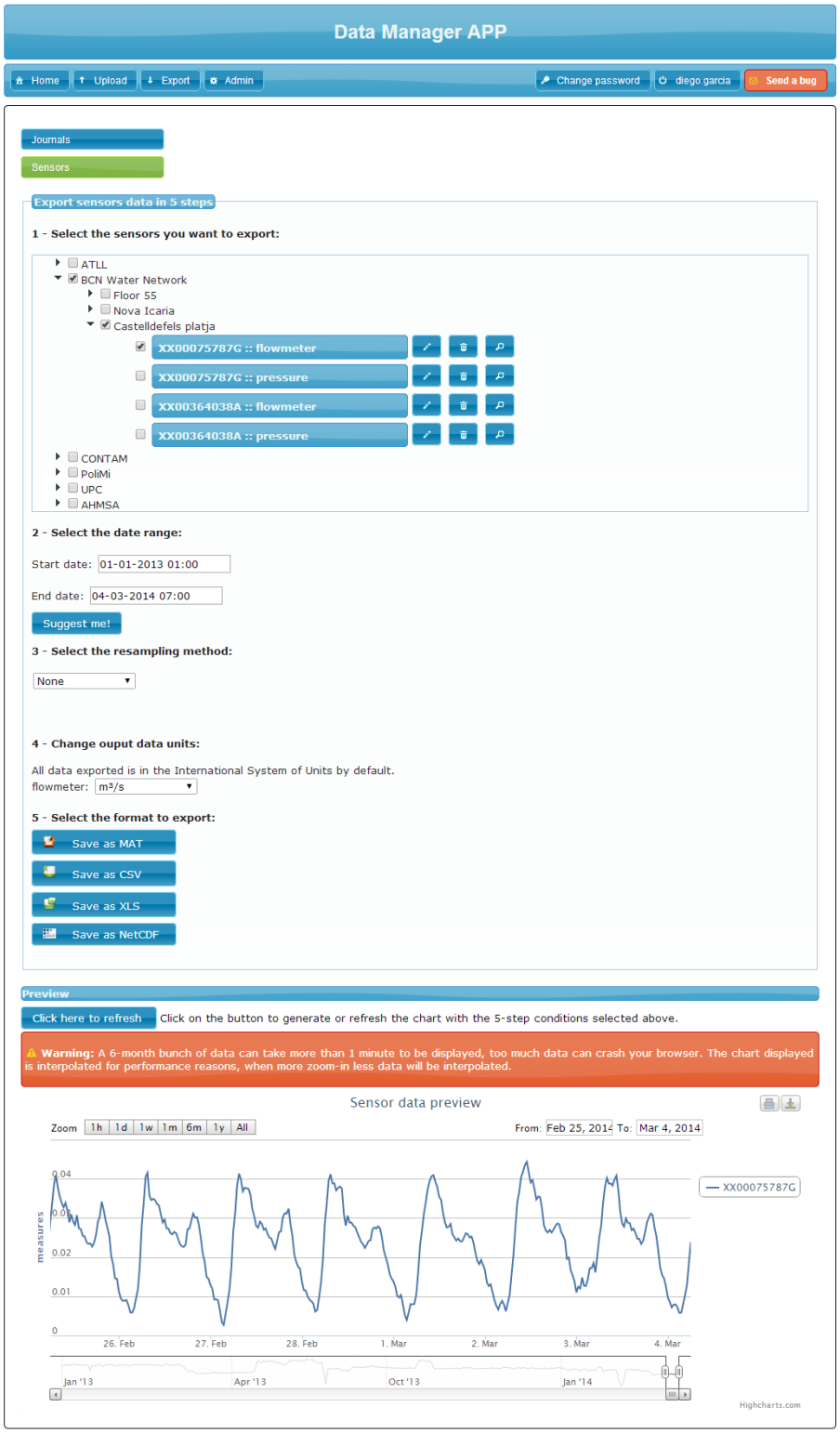

Figure 5. Data Management Web Application screenshot

is a widely used numerical computing and programming platform in many research institutions and industrial enterprises, which makes it a convenient prototyping and development framework. This tool includes a Graphical User Interface (GUI) to configure different modules and to run the validation and reconstruction processes with the configured settings (Figure 6). This GUI is composed by six panels. Following the numeration in the figure, each panel has the following purpose:

1. **Input data.** The user can select the *.mat* file path in SAC format and load the data into the tool.
2. **Signals list.** This panel shows the listing of the signals loaded in the previous panel.
3. **Fault generator.** This module provides a fault generator framework in order to simulate different types of fault, thus the user can apply a fault to the selected signals. The faults available are: freezing, offset, drift, noise and communication.
4. **Data ranges.** This panel allows the user to indicate the season periodicity L (cycle time) of the TSM. For instance, if a signal shows a daily pattern, the cycle time is 24 hours (86400 seconds). The user can also define the number of identification and validation cycles. The rest of the data will be used as testing dataset.
5. **Tests and Models.** This panel lists the tests and the models available. Here, the user can select the tests to apply and configure the required parameters, depending on the models and tests selected.
6. **Output and Reporting.** In this panel, the user can enter the path where the results will be recorded and select the reporting options.

Algorithm 1 Data validation

Require: $y_{raw}(k)$
  $v(k) = 1$; # Initialise $v(k)$
  $q(k) = 0$; # Initialise $q(k)$
  for all Validation levels $n = 0, \cdots, 5$ **do**
    Check validation level test n;
    if Validation test n passed **then**
      $l_n(k) = 1$; # Set level n as passed
      $q(k) = q(k) + 1$; # Increase quality level of datum $y_{raw}(k)$
    else
      $l_n(k) = 0$; # Set level n as not passed
      $v(k) = 0$; # Void datum $y_{raw}(k)$
    end if
  end for
  if $v(k) = 1$ **then**
    $y_{val}(k) = y_{raw}(k)$; # Datum $y_{raw}(k)$ is validated
  else
    $y_{val}(k) = []$; # Datum $y_{raw}(k)$ is voided
  end if
  return $v(k)$, $\mathbf{l}(k)$, $q(k)$, $y_{val}(k)$

The input dataset selected in the *Input data* panel (Figure 6) is divided in three different subsets (i.e. calibration, validation and testing) in order to calibrate and validate the models and parameters, and check the sensors' raw data, respectively. The use of different data subsets allows the analysis and validation of how these models will generalize to an independent dataset. Calibration and validation subsets are assumed to be faultless, whilst testing dataset includes the faulty scenario to be diagnosed. The different subset ranges are defined by the user according to the parameters entered in the *Data ranges* panel (Figure 6).

Once all the required parameters are set by the user the process may be started, which will sequentially apply the presented methodology to the data. This process is divided in three different stages, namely Calibration, Validation and Reconstruction, respectively. First, the Calibration stage is executed using the calibration and validation datasets in order to learn and estimate the parameters required by the tests and the models to be applied (see Sections 2 and 3). Once the models and the tests are calibrated, the Validation stage runs the sequence of tests in order to validate the

Algorithm 2 Data reconstruction

Require: $y_{raw}(k)$, $v(k)$
  if $v(k) = 0$ **then**
    Compute $MSE_{SM}(k)$ and $MSE_{TSM}(k)$; # Evaluate MSE for each model
    if $MSE_{SM}(k) < MSE_{TSM}(k)$ **then**
      $y_{rec}(k) = \hat{x}_{SM}(k)$; # Reconstructed datum $y_{raw}(k)$ is given by SM estimation
    else
      $y_{rec}(k) = \hat{x}_{TSM}(k)$; # Reconstructed datum $y_{raw}(k)$ is given by TSM estimation
    end if
  else
    $y_{rec}(k) = []$; # If datum $y_{raw}(k)$ is validated, no reconstruction is needed
  end if
  return $y_{rec}(k)$

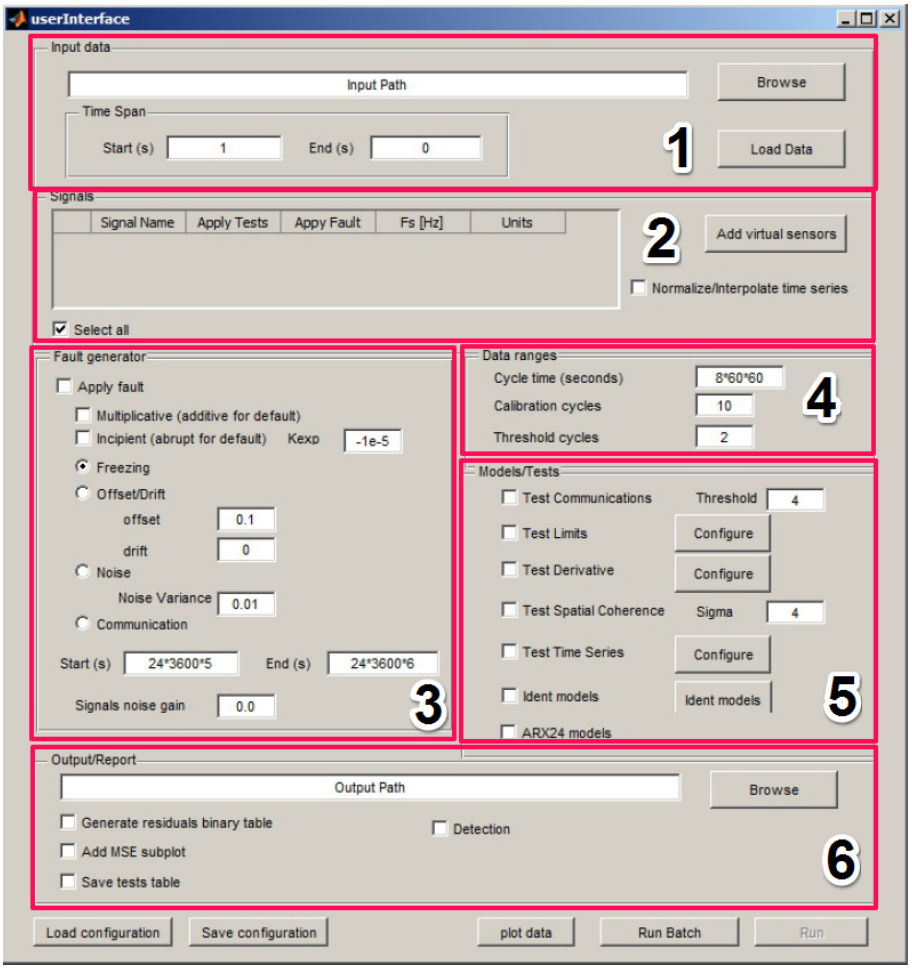

Figure 6. Validation and Reconstruction Matlab Tool

testing dataset (see Section 2). Each test applied labels each datum $y_{raw}(k)$ with a flag (**l** in Figure 2 and Algorithm 1) to indicate whether the test has been fulfilled. Finally, in the Reconstruction stage (see Section 3.3), the model with best performance (i.e. lowest MSE) is selected in order to replace the invalidated datum at the Validation stage (datum with $v = 0$ in Figure 2 and Algorithm 1) by its corresponding reconstructed estimation. In Figure 7, the data flow between these three stages is presented.

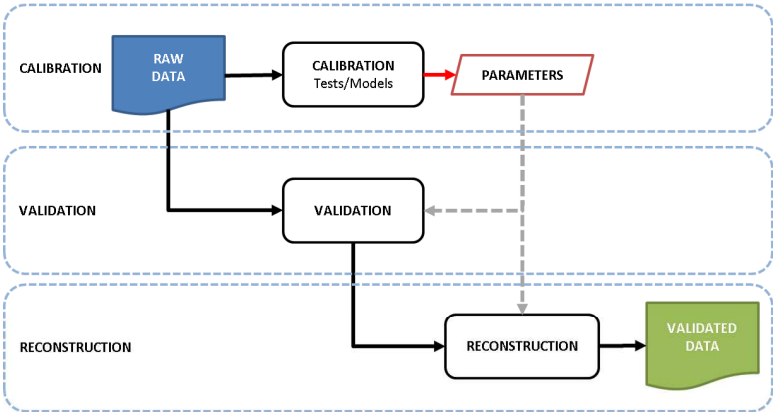

Figure 7. Validation and Reconstruction data flow diagram

5. Case Study: Catalonia Regional Water Network

5.1. Description

The Catalonia regional water network managed by ATLL company (Figure 8) supplies water to the metropolitan area of Barcelona. Most of the population of the region (approximately 4.5 million people) is concentrated in this area. This network transports the drinking water from the main water treatment plants (ETAPs), which take the water from two different rivers (Llobregat and Ter), towards the main storing and buffer tanks of 116 different municipalities in the Barcelona metropolitan area, using about 1045 km of pipes of up to 3 m diameter. The network is composed by 170 storage tanks, 67 pumps and 212 demand sectors, and is monitored using more than 200 flow meters and 115 tank level sensors by means of a SCADA system with 10 minutes sample time.

5.2. Results

In this section, some results obtained with the methodology introduced here are presented, using the tool introduced in Section 4. These results are based on a variety of real situations in order to show the performance of the methodology and the tool presented. The dataset used to obtain these results is the network's raw data collected by ATLL company, including flow meter measurements, level meter measurements, valve positions and communication system alarms.

In Figure 9, the first fault scenario considered is shown. The top plot shows the measured signal (solid black line) gathered from the flow meter D6FT00204_CI, with a time range from 3rd to 8th of January 2014. On the one hand, the pattern of the measured signal for days 4th, 6th and 7th of January, respectively, present a similar behavior, with around 300 m^3/h peak. On the other hand, the pattern of the measured signal on January the 5th presents negative pumping flows, which should be corrected. This change in the pattern is detected by the physical range test (Level 1

Figure 8. ATLL's Catalonia Regional Water Network

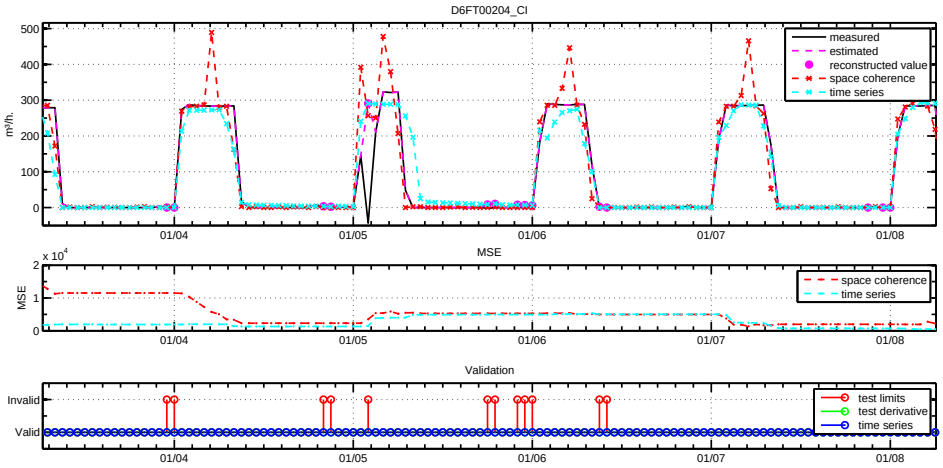

Figure 9. Results of the validation and reconstruction methodology, flow meter D6FT00204_CI

in Figure 2). The detection is indicated by the flags in the bottom plot in Figure 9: if the test flag is set to zero (*Valid* state) the datum pass the test; if it is set to one (*Invalid* state) the datum is invalidated. Using these flags, it may be noted how negative flows are detected e.g. at the beginning of days 4th, 5th and 6th of January, although their magnitude are too low to make them visible in the top plot in Figure 9. The top plot in the latter figure also shows the SM (dashed red line) and the HW TSM (dashed cyan line) estimations. The invalid observations are replaced by estimations (magenta dots) obtained from the model having the best performance according to their MSE (middle plot in Figure 9), as introduced in Section 3.3.

Figure 10 presents a different scenario, where the measured flow signal E6FT00102_CI exhibits a peak of high magnitude on the fifth day of January 2014. In contrast to the previous scenario, in the current scenario there is only the TSM model available, since no SM model can be obtained due to the topological configuration of the network. However, validation and reconstruction can also be performed since TSM only needs historical records from the single sensor under study to operate, i.e. does not need additional data gathered by other related sensors as is the case with SM models. The peak appearing in the top plot in Figure 10 is detected by the physical range test and the trend test, respectively (Level 1 and Level 2 in Figure 2). The detection is indicated in the bottom plot in Figure 10, and reconstructed by the HW TSM model (magenta dots, top plot in Figure 10).

An additional scenario is presented in Figure 11. Here, the network topological configuration allows a SM model to be obtained, using the spatial relation of the sensors involved as presented in Section 3.1. The top plot in Figure 11 shows the measured flow signal D6FT00204_CI (solid black line). This scenario exhibits a communication fault affecting only the sensor under study for a period of three days, between t = 880 h and t = 952 h. The measured data when the fault is occurring are also available (*Original data*, dash-dotted line in Figure 11) and are used to check the performance of the data reconstruction model utilised. Also, the threshold boundaries in (7) for each model are also depicted (red and blue dotted lines for SM and TSM, respectively). The communication problem is detected by the Level 0 test in Figure 2 and the missing data over the faulty period are reconstructed by the model exhibiting the best performance according to their MSE (bottom subplot in Figure 11), i.e. the TSM model in this particular case.

In Figure 12, a scenario involving the flow meter E6FT00502 CI is presented. In this case, a general communication fault affects all the sensors, a common situation occurring in actual water monitoring systems when e.g. the concentrator (a device collecting data from sensors installed in a particular zone) drops. Similarly as in the scenario in Figure 11, a SM model is available using the corresponding spatially related sensors data. However, in this particular case the rest of the sensors involved in the SM model (i.e. flow meter E6FT00502_CI in Figure 12, flow meter D6FT00201_CI in Figure 13) are all affected by the same communication fault, hence they are not available for data reconstruction after the communication fault occurs and, consequently, the SM model can not be considered in this

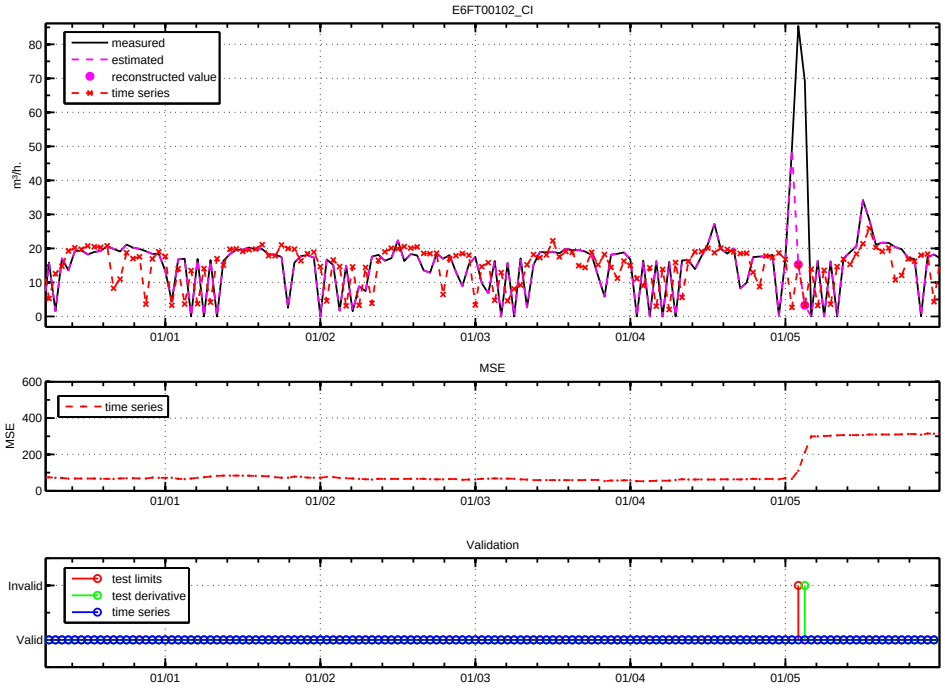

Figure 10. Results of the validation and reconstruction methodology on the flow meter E6FT00102_CI

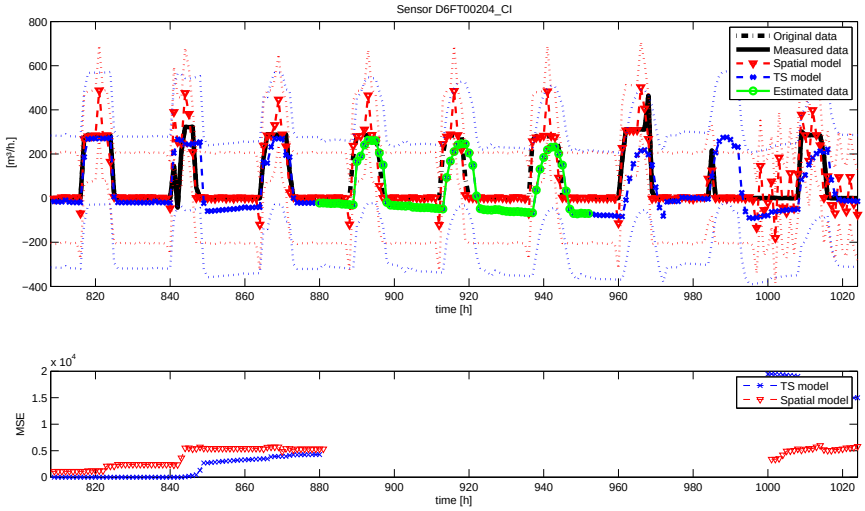

Figure 11. Results of the validation and reconstruction methodology, flow meter D6FT00204_CI

case due to the lack of information. Again, the only available model for reconstruction is the TSM (similarly as in the scenario in Figure 10) which is finally used for the missing data reconstruction in the scenario considered in Figure 12, based on the limited available information in this particular case.

Finally, two different scenarios involving the flow meter E6FT00502_CI are considered. On the one hand, in the scenario in Figure 14, a communication fault affecting the corresponding flow meter is presented, which does not transmit data in one day period (from $t = 1024$ h to $t = 1048$ h). In this particular case, the communication fault only affects the latter sensor, hence the corresponding SM is available because the spatially related sensors (e.g. D6FT00201_CI) are not affected by this fault. In this scenario, the SM model is used for missing data reconstruction, since it performs better than the corresponding HW TSM model (bottom subplot in Figure 14). It may be noted that the use of the SM assumes that the model input sensor measurement is faultless when the SM is used for data reconstruction. This may be assured since the input model integrity is checked by the methodology presented here at its corresponding stage and, if not verified, the validation test at this stage is not fulfilled.

On the other hand, an offset fault of magnitude 25 % full scale affecting the flow meter E6FT00502_CI, also common in this kind of sensors, is presented in Figure 15, lasting for three days (from $t = 1024$ h to $t = 1096$ h). As in the previous scenario, the SM model performs better than HW TSM before the fault is produced (see Figure 15 bottom subplot) and hence it is used for invalid data estimation. In this particular scenario, it may be noted how the measured signal is out of the SM threshold boundaries (red dotted line) for the whole fault scenario, whilst it remains bounded by the HW TSM threshold boundaries (blue dotted line) for part of the first day after the fault is produced. This behavior is due to the adaptation of the HW TSM to the input signal, as its estimation depends on the historic records of the measurements, as detailed in Section 3.1. Hence, it should be considered that, when used for data validation, the prognosis derived from the application of the time series consistency test will expire after a certain time after the fault is produced, when using measurement historic records.

6. Conclusions

In this paper, a data validation and reconstruction methodology is introduced to overcome the sensor problems arising in CIS, such as water networks. The validation strategy is based on a set of data quality tests that allow to detect potentially erroneous data. Then, a reconstruction scheme is defined using SM and TSM to provide an estimation based on the model having the best fit, also providing prediction intervals for the forecasted reconstructed data. In addition, a software tool is described to provide a homogeneous and accessible database by a user-friendly interface, and to apply the methodology presented here. Finally, some results obtained using data from a real network

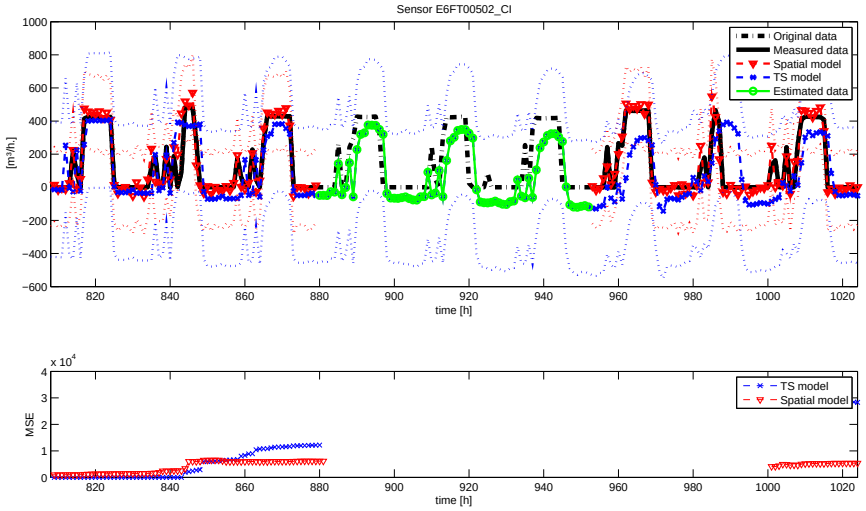

Figure 12. Results of the validation and reconstruction methodology, flow meter E6FT00502_CI

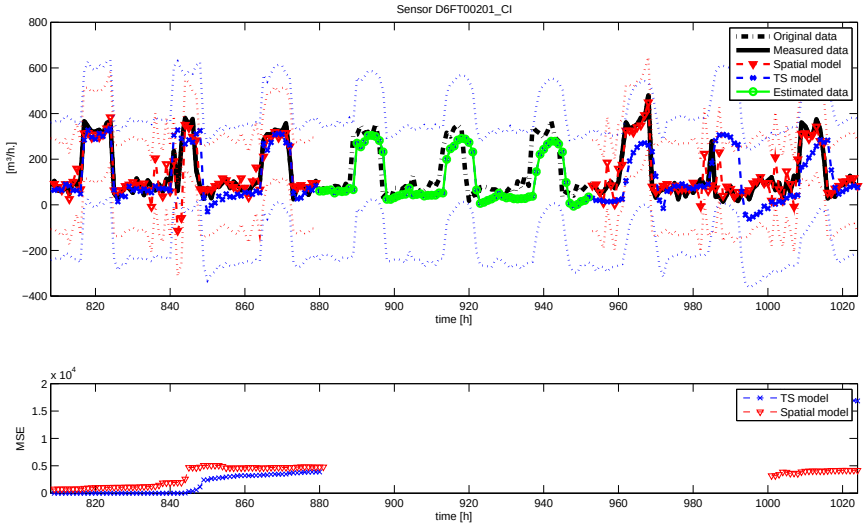

Figure 13. Results of the validation and reconstruction methodology, flow meter D6FT00201_CI

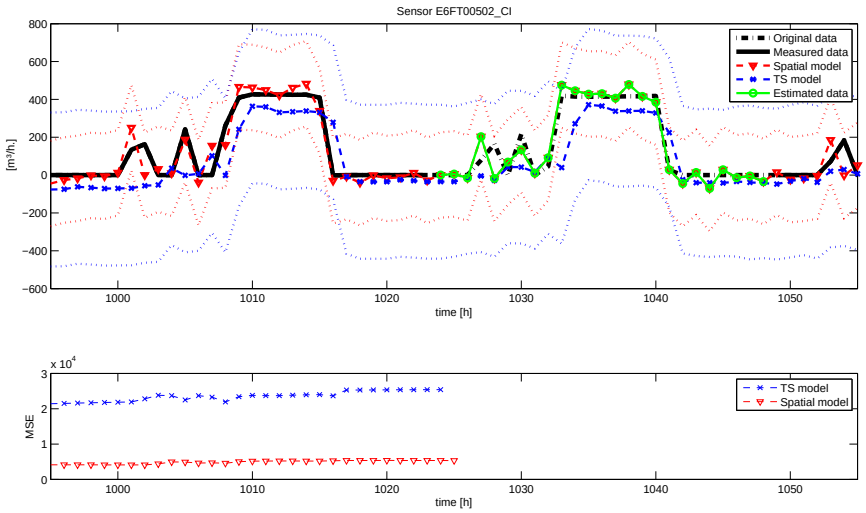

Figure 14. Results of the validation and reconstruction methodology on the flow meter E6FT00502_CI

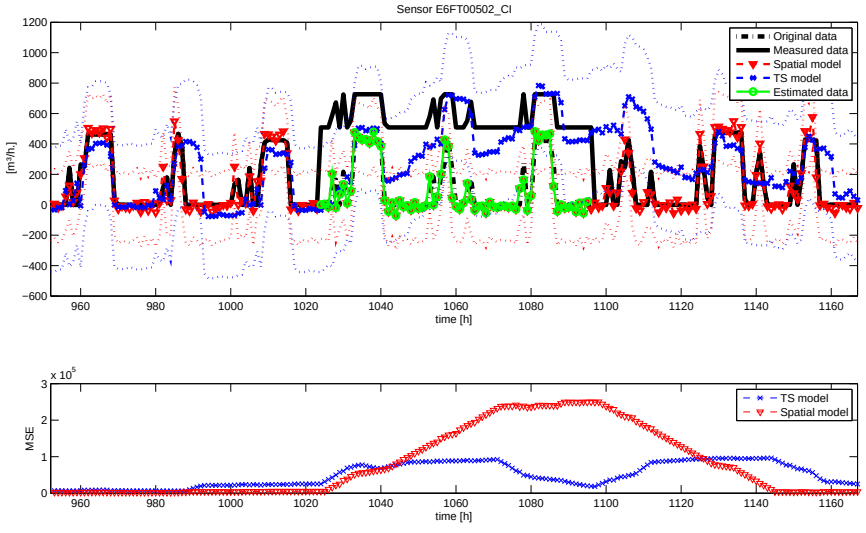

Figure 15. Results of the validation and reconstruction methodology on the flow meter E6FT00502_CI

located in the Catalonia area are presented using the software described, showing the ability of the methodology to detect and reconstruct anomalous data. In future steps of this work, the proposed methodology and tool are going to be applied to the whole Catalonia Regional network, since in the latter network not only the hydraulic sensors considered here are monitored, but also e.g. the water quality sensors.

Acknowledgement

This work has been partially funded by the Spanish Ministry of Science and Technology through the Project ECOCIS (Ref. DPI2013-48243-C2-1-R) and Project HARCRICS (Ref. DPI2014-58104-R), and by EFFINET grant FP7-ICT-2012-318556 of the European Commission.

References

[1] M. Schtze, A. Campisano, H. Colas, W. Schilling, P. A. Vanrolleghem, Real time control of urban wastewater systems - where do we stand today?, Journal of Hydrology 299 (2004) 335–348. doi:10.1016/j.jhydrol.2004.08.010.

[2] M. Marinaki, M.; Papageorgiou, Optimal Real-time Control of Sewer Networks, Springer, 2005.

[3] V.K.Kanakoudis, D.K.Tolikas, The role of leaks and breaks in water networks: technical and economical solutions, Water Supply: Research and Technology-Aqua 50 (5) (2001) 301–311.

[4] V.K.Kanakoudis, S. Tsitsifli, Water pipe network reliability assessment using the dac method, Desalination and Water Treatment 33 (1-3) (2011) 97–106.

[5] S.Tsitsifli, V.K.Kanakoudis, I. Bakouros, Pipe networks risk assessment based on survival analysis, Water Resources Management 25 (14) (2011) 3729–3746.

[6] J. Quevedo, V. Puig, G. Cembrano, J. Blanch, J. Aguilar, D. Saporta, G. Benito, M. Hedo, A. Molina, Validation and reconstruction of flow meter data in the Barcelona water distribution network, Control Engineering Practice 18 (6) (2010) 640–651. doi:10.1016/j.conengprac.2010.03.003.

[7] R. Pérez, G. Sanz, V. Puig, J. Quevedo, M. A. Cugueró-Escofet, F. Nejjari, J. Meseguer, G. Cembrano, J. M. M. Tur, R. Sarrate, Leak Localization in Water Networks. A Model-Based Methodology Using Pressure Sensors Applied to a Real Network in Barcelona, IEEE Control Systems Magazine 34 (2014) 24–36. doi:10.1109/MCS.2014.2320336.

[8] J. Quevedo, M. A. Cugueró, R. Pérez, F. Nejjari, V. Puig, J. M. Mirats, Leakage location in water distribution networks based on correlation measurement of pressure sensors, in: 8th IWA Symposium on System Analysis and Integrated Assessment (WATERMATEX 2011), San Sebastian, 2011.

[9] R. Pérez, V. Puig, J. Pascual, J. Quevedo, E. Landeros, A. Peralta, Methodology for leakage isolation using pressure sensitivity analysis in water distribution networks, Control Engineering Practice 19 (10) (2011) 1157–1167. doi:10.1016/j.conengprac.2011.06.004.

[10] M. A. Cugueró-Escofet, M. Christodoulou, J. Quevedo, V. Puig-Cayuela, D. García, M. Michaelides, Combining Contaminant Event Diagnosis with Data Validation / Reconstruction : Application to Smart Buildings, in: In Proceedings of IEEE 22nd Mediterranean Conference on Control and Automation (MED14), Palermo, Italy, 2014, pp. 293–298. doi:10.1109/MED.2014.6961386.

[11] M. A. Cugueró, J. Quevedo, V. Puig, D. García, Inconsistent sensor data detection/correction: Application to environmental systems, in: Neural Networks (IJCNN), 2014 International Joint Conference on, 2014, pp. 84–90.

[12] S. Narasimhan, C. Jordache, Data Reconciliation and Gross Error Detection: an Intelligent Use of Process Data, Gulf Professional Publishing, 1999. doi:10.1016/B978-088415255-2/50018-5.

[13] J. A. Romagnoli, M. C. Sánchez, Data Processing and Reconciliation for Chemical Process Operations, Vol. 2 of Process Systems Engineering, Elsevier, 1999. doi:10.1016/S1874-5970(00)80015-6.

[14] M. Fagiani, S. Squartini, L. Gabrielli, S. Spinsante, F. Piazza, A Review of Datasets and Load Forecasting Techniques for Smart Natural Gas and Water Grids: Analysis and Experiments, Neurocomputing 170 (October) (2015) 448–465. doi:10.1016/j.neucom.2015.04.098.

[15] H. Jorgensen, S. Rosenrn, H. Madsen, P. Mikkelsen, Quality control of rain data used for urban runoff systems, Water Science and Technology 37 (11) (1998) 113–120.
URL http://www.sciencedirect.com/science/article/pii/S0273122398003230

[16] M. Mourad, J.-L. Bertrand-Krajewski, A method for automatic validation of long time series of data in urban hydrology, Water Science & Technology 45 (4-5) (2002) 263–270.

[17] V. Venkatasubramanian, R. Rengaswamy, K. Yin, S. N. Kavuri, A review of process fault detection and diagnosis: Part i: Quantitative model-based methods, Computers & Chemical Engineering 27 (3) (2003) 293–311. doi:10.1016/S0098-1354(02)00160-6.

[18] N. Branisavljević, Z. Kapelan, D. Prodanović, Improved real-time data anomaly detection using context classification, Journal of Hydroinformatics 13 (2011) 307. doi:10.2166/hydro.2011.042.

[19] G. Lempio, C. Podlasly, T. Einfalt, NIKLAS - Automatical quality control of time series data (2000) (2010) 1–6.

[20] F. Edthofer, J. Van Den Broeke, J. Ettl, W. Lettl, a. Weingartner, Reliable online water quality monitoring as basis for fault tolerant control, Conference on Control and Fault-Tolerant Systems, SysTol'10 - Final Program and Book of Abstracts (2010) 57–62doi:10.1109/SYSTOL.2010.5675985.

[21] K. Tsang, Sensor data validation using gray models, ISA Transactions 42 (2003) 9–17.

[22] D. Burnell, Auto-validation of district meter data, in: CCWI '03 Advances in Water Supply Management, London, 2003.

[23] S. Makridakis, S. Wheelwright, R. Hyndman, Forecasting methods and applications, John Wiley & Sons, 1998.

[24] J. Quevedo, J. Blanch, V. Puig, J. Saludes, S. Espin, J. Roquet, Methodology of a data validation and reconstructions tool to improve the reliability of the water network supervision, in: International Conference of IWA Water Loss, Sao Paolo, Brazil, 2010.

[25] P. R. Winters, Forecasting sales by exponentially weighted moving averages, Management Science 6 (52) (1960) 324–342.
[26] R. G. Brown, Statistical Forecasting for Inventory Control, New York: McGraw-Hil, 1959.
[27] J. Taylor, Short-term electricity demand forecasting using double seasonal exponential smoothing, The Journal of the Operational Research Society 54 (8) (2003) 799–805.
[28] J. W. Taylor, Triple seasonal methods for short-term electricity demand forecasting, European Journal of Operational Research 204 (1) (2010) 139–152. doi:10.1016/j.ejor.2009.10.003.
URL http://linkinghub.elsevier.com/retrieve/pii/S037722170900705X
[29] C. Pegels, Exponential forecasting: Some new variations, Management Science 15 (1969) 311–315.
[30] E. S. Gardner Jr., Exponential Smoothing: The State of the Art - Part II, International Journal of Forecasting 22 (4) (2006) 637–666.
URL http://www.sciencedirect.com/science/article/pii/S0169207006000392
[31] S. Gelper, R. Fried, C. Croux, Robust Forecasting with Exponential and Holt-Winters Smoothing, Journal of forecasting 29 (June 2009) (2010) 285–300. doi:10.1002/for.
URL http://onlinelibrary.wiley.com/doi/10.1002/for.1125/abstract
[32] M. Basseville, I. Nikiforov, Detection of Abrupt Changes: Theory and Application, Prentice-Hall, Inc., 1993.
[33] V. Puig, Fault diagnosis and fault tolerant control using set-membership approaches: Application to real case studies, International Journal of Applied Mathematics and Computer Science 20 (4) (2010) 619–635.
[34] S. X. Ding, Model-based Fault Diagnosis Techniques, Springer, 2008.